Avrupa Bilim ve Teknoloji Dergisi
Sayı 31 (Ek Sayı 1), S. 461-468, Aralık 2021

Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
No. 31 (Supp. 1), pp. 461-468, December 2021

Research Article

A Benchmark for Feature-injection Architectures in Image Captioning

Rumeysa Keskin[1], Özkan Çaylı [1], Özge Taylan Moral [1*], Volkan Kılıç[1], Aytuğ Onan[2]

[1] İzmir Katip Celebi University, Faculty of Engineering and Architecture, Department of Electrical and Electronics, İzmir, Turkey, (ORCID: 0000-0001-8452-8221, 0000-0002-3389-3867, 0000-0003-0482-267X, 0000-0002-3164-1981), 160403043@ogr.ikcu.edu.tr, y200207004@ogr.ikcu.edu.tr, ozgetaylan.moral@ikcu.edu.tr, volkan.kilic@ikcu.edu.tr

[2] İzmir Katip Celebi University, Faculty of Engineering and Architecture, Department of Computer Engineering, İzmir, Turkey, (ORCID: 0000-0002-9434-5880), aytug.onan@ikcu.edu.tr

(First received 22 October 2021 and in final form 6 December 2021)

(**DOI:** 10.31590/ejosat.1013329)

ATIF/REFERENCE: Keskin, R., Çaylı, Ö., Moral, Ö.T., Kılıç, V., & Onan, A. (2021). A Benchmark for Feature-injection Architectures in Image Captioning. *European Journal of Science and Technology*, (31), 461-468.

Abstract

Describing an image with a grammatically and semantically correct sentence, known as image captioning, has been improved significantly with recent advances in computer vision (CV) and natural language processing (NLP) communities. The integration of these communities leads to the development of feature-injection architectures, which define how extracted features are used in captioning. In this paper, a benchmark of feature-injection architectures that utilize CV and NLP techniques is reported for encoder-decoder based captioning. Benchmark evaluations include Inception-v3 convolutional neural network to extract image features in the encoder while the feature-injection architectures such as init-inject, pre-inject, par-inject and merge are applied with a multi-layer gated recurrent unit (GRU) to generate captions in the decoder. Architectures have been evaluated extensively on the MSCOCO dataset across eight performance metrics. It has been concluded that the init-inject architecture with 3-layer GRU outperforms the other architectures in terms of captioning accuracy.

Keywords: Convolutional Neural Network, Feature-injection Architectures, Gated Recurrent Unit.

Görüntü Altyazılamada Öznitelik Enjeksiyon Mimarileri için Bir Kıyaslama

Öz

Görüntü altyazılama olarak bilinen, bir görüntüyü dilbilgisel ve anlamsal olarak doğru bir cümle olarak tanımlama, bilgisayarlı görme ve doğal dil işleme alanlarındaki son gelişmelerle birlikte önemli ölçüde ilerlemiştir. Bu iki alanın birleştirilmesi, çıkarılan özniteliklerin altyazı oluşturmada nasıl kullanılacağını tanımlayan öznitelik enjeksiyon mimarisinin geliştirilmesine öncülük etmiştir. Bu çalışmada, bilgisayarlı görme ve doğal dil işleme tekniklerini kodlayıcı-kod çözücü tabanlı görüntü altyazılamada kullanan öznitelik enjeksiyon mimarilerinin bir karşılaştırılması raporlanmaktadır. Kıyaslama değerlendirmelerinde, Inception-v3 evrişimsel sinir ağı, kodlayıcıda görüntü özniteliklerini çıkarmak için kullanılırken; init-inject, pre-inject, par-inject ve merge gibi öznitelik enjeksiyon mimarileri altyazı üretmek için çok katmanlı kapılı tekrarlayan birim ile kod çözücüde uygulanmaktadır. Mimariler sekiz performans metriği ile MSCOCO veri kümesi üzerinde kapsamlı bir şekilde değerlendirilmiştir. 3 katmanlı GRU ile init-inject mimarisinin altyazı doğruluğu açısından diğer mimarilerden daha iyi performans gösterdiği sonucuna varılmıştır.

Anahtar Kelimeler: Evrişimsel Sinir Ağları, Öznitelik Enjeksiyon Mimarileri, Kapılı Tekrarlayan Birim.

[1] Corresponding Author: ozgetaylan.moral@ikcu.edu.tr

1. Introduction

Captioning is an automated image description with a meaningful and grammatically correct sentence that has applications in many areas, such as description generation in social media (Chiarella, Yarbrough, & Jackson, 2020), image indexing (Chang, 1995), and assistance for the visually impaired (Baran, Moral, & Kılıç, 2021; Çaylı, Makav, Kılıç, & Onan, 2020; Keskin, Moral, Kılıç, & Onan, 2021; Makav & Kılıç, 2019b). This task is accomplished using a combination of CV and NLP techniques.

Earlier studies for image captioning have dealt with different approaches: retrieval-based, template-based and encoder-decoder based methods. The retrieval-based approach, which is a traditional image captioning approach, retrieves reference caption of images from a dataset and analyzes its semantic information for generating a caption of a new image based on similarities (X. Liu, Xu, & Wang, 2019). Therefore, in this approach, caption generation is limited to the properties of the dataset and cannot generate novel descriptions which are not in training (Ordonez, Kulkarni, & Berg, 2011). Researches in image captioning have advanced remarkably with the template-based approach, which utilizes object detection and language models (Kulkarni et al., 2013). This approach detects objects, attributes, and spatial relationships from images to generate a syntactically correct sentence from a template that is a set of most likely words. This method improves the generated captions since it precisely complies with the grammatical rules. However, the generated sentence is not comparable with human style as it is only similar to the template, and it cannot add new words or reorder them.

Effective image captioning methods have been introduced with the development of encoder-decoder based approaches. Unlike retrieval-based and template-based, this approach combines a convolutional neural network (CNN) and a recurrent neural network (RNN) to describe images (Makav & Kılıç, 2019a). A CNN based encoder is employed to extract image features, then the features have been fed to the decoder. There are many types of research devoted to performance improvement in CNNs, resulting in the emergence of advanced CNN architectures such as Inception-v3 (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016), Xception (Chollet, 2017), and ResNet (Targ, Almeida, & Lyman, 2016) that are widely used in encoder design. Image features extracted by CNN-based encoders are processed in RNN-based decoders to generate natural language captions using this information. However, the simple RNN has issues like the vanishing and exploding gradient problems that prevent modeling of long-term dependencies (Ouyang, Zeng, Li, & Luo, 2020). To overcome these issues, advanced RNNs are introduced, including the long-short term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent unit (GRU) (Chung, Gulcehre, Cho, & Bengio, 2014). LSTM network solves the vanishing gradient and losing the content of the information problem with the addition of memory cells to capture long-term dependencies.

GRU controls the information flow without additional memory cells and uses fewer gates that correspond to only one-fourth parameters of the LSTM (Gao & Glowacka, 2016). In this regard, GRU has been proven to be an efficient choice over traditional RNNs or LSTM due to the high accuracy with the lower computation (Gers & Schmidhuber, 2001). RNN-based decoders require a feature-injection that utilizes the image and linguistic features to generate a more meaningful caption (Baran et al., 2021; Çaylı et al., 2020), (Devlin et al., 2015; Tanti, Gatt, & Camilleri, 2018; Vinyals, Toshev, Bengio, & Erhan, 2016). The feature-injection architectures could be categorized into init-inject, pre-inject, par-inject, and merge (Tanti et al., 2018). The image feature is fed to the initial hidden state of the RNN in the init- inject (Devlin et al., 2015), (S. Liu, Zhu, Ye, Guadarrama, & Murphy, 2016), while the first input of the RNN is fed by the image feature as in pre-inject architecture (Nina & Rodriguez, 2015; Vinyals, Toshev, Bengio, & Erhan, 2015). The image feature is used with the linguistic feature in parallel as an input to the RNN in par-inject architecture (Donahue et al., 2015; Yao, Pan, Li, Qiu, & Mei, 2017). The image feature is exposed to the system after the GRU processes in merge architecture (Baran et al., 2021; Mao et al., 2015).

In this study, a benchmark of feature-injection architectures, i.e., init-inject, par-inject, pre-inject and merge, is reported for image captioning based on the encoder-decoder approach. The Inception-v3 architecture is used for all experiments to extract image features due to its high-level performance in ILSVRC 2015 (He, Zhang, Ren, & Sun, 2016). GRU is employed in the decoder to generate image captions because of its computational efficiency and prediction accuracy. As it is reported in (Tanti et al., 2018), a fully connected (FC) layer optimized the captioning accuracy in par-inject architecture. Based on this conclusion, benchmark evaluations have been extended for all architectures, including a connection of an FC layer to the decoder. In addition, multi-layer GRU is employed to examine the effect of layer size in caption generation. The motivation behind the increasing layer size is to enhance the memory ability of the model to compute more complex representations in learning sequential data to provide a captioning model that generates more accurate predictions (Keskin et al., 2021; Kılıç, 2021; Tao, Wang, Sánchez, Yang, & Bai, 2019). The experiments were evaluated on the MSCOCO dataset (T.-Y. Lin et al., 2014) with commonly used performance metrics BLEU-n (n = 1, 2, 3, 4) (Papineni, Roukos, Ward, & Zhu, 2002), ROUGE-L (C.-Y. Lin, 2004), SPICE (Anderson, Fernando, Johnson, & Gould, 2016), METEOR (Banerjee & Lavie, 2005) and CIDEr (Vedantam, Lawrence Zitnick, & Parikh, 2015).

The rest of the paper is organized as follows: Section 2 covers the encoder-decoder based approach and feature-injection architectures with theoretical foundations. Section 3 introduces the dataset, performance metrics and results with implementation details. Conclusions are drawn in Section 4.

2. Image Captioning Methods

The feature-injection architectures based on the encoder-decoder approach to utilize the image and linguistic features are described in this section.

2.1. Encoder-Decoder Based Approach

In encoder-decoder based approach, it is intended to maximize the probability $p(S|I)$ for generating the best descriptions (X. Liu et al., 2019) as follows

$$\theta^* = \arg\max_{\theta} \sum_{(I,S)} \log p(S|I;\theta) \quad (1)$$

where θ represents learning parameter, I is the input image and $S = \{S_0, S_1, \dots, S_{t-1}\}$ is the corresponding caption. Since the varying length of the caption is generated for each image, the probability calculation is expressed by the chain rule (X. Liu et al., 2019),

$$\log p(S|I) = \sum_{t=0}^{N} \log p(S_t|I, S_0, \dots, S_{t-1}) \quad (2)$$

where N is the caption length. The conditional probability $p(S_t|I, S_0, \dots, S_{t-1})$ in Eq. 2 is modeled with RNN taking the input at the current time step t and the output of the previous time step $t-1$. This recursive process predicts every word in the caption, which produces a probability distribution over all possible words according to the current word and context from previous time steps.

Encoder - CNN: The encoder is the first step of the encoder-decoder based approach that extracts the image features using a CNN architecture. The Inception-v3, which consists of 48 convolutional, pooling, and FC layers with an input image size of 3×299×299, was utilized to extract image features in this study. The process of the input image was propagated until the global average pooling layer of the Inception-v3, resulting in a feature vector with the length of 2048 to feed into the decoder for the comparison of feature-injection architectures.

Decoder - GRU: The next step is the decoder, where the feature-injection architectures are applied to generate a caption word-by-word using both image and linguistic features. The multi-layer GRU based decoder mainly consists of the FC layer, embedding layer and GRUs, which is illustrated in Figure 1. The FC layer utilizes an activation function on input with weight to calculate the output. This layer normalizes the output of the GRU with a logarithmic softmax function that improves loss calculation. Here, this layer is first utilized to produce an image feature vector with the reduced size before the GRU layer to test the feature-injection architectures. Meanwhile, the embedding layer processes tokens representing numerical components and generates an embedding vector (or word embedding) containing linguistic features to feed the GRU, which is a type of RNN with a gating mechanism to control the information flow through cells. GRU consists of a hidden state vector, update and reset gates. The flow of information on GRU is as follows (Chung et al., 2014):

$$r_t = \sigma(W_r x_t + W_r h_{t-1}) \quad (3)$$

$$z_t = \sigma(W_z x_t + W_z h_{t-1}) \quad (4)$$

$$u_t = tanh(W_h x_t + W_h(r_t \odot h_{t-1})) \quad (5)$$

$$h_t = (1 - z_t)h_{t-1} + z_t u_t \quad (6)$$

where x_t and h_t are the input and the hidden state vectors, r_t, z_t and u_t corresponds to the reset gate, update gate and candidate hidden vector, respectively. W denotes weight matrices, σ and *tanh* are sigmoid and hyperbolic tangent. $\odot$ denotes the element-wise multiplication operator. The multi-layer GRU is a combination of K-GRU for $k = 1, \dots, K$. The first GRU layer takes the embedding vector, which is generated using start-token from the embedding layer. The output vector of the first layer feeds to the next GRU layer, and this process is continued K-times reaching the last output is generated, which is the input for the FC layer. The FC layer generates the first token, which is computed by the embedding layer in the next time step. The procedure is repeated T-times to reach the end token. All generated tokens are converted to the image caption. Inject-based and merge architectures are applied to the multi-layer GRU based decoder to see the effect of layer size on generating caption.

2.2. Feature-Injection Architectures

Images can be incorporated into the decoder with feature-injection architectures in two different ways (i.e., inject-based and merge) using a fixed-length image feature vector and linguistic feature vector (embedding vector) from the encoder and embedding layer, respectively. The inject-based architecture is designed to utilize both image feature and linguistic feature vector to the decoder, such as init-inject, pre-inject and par-inject.

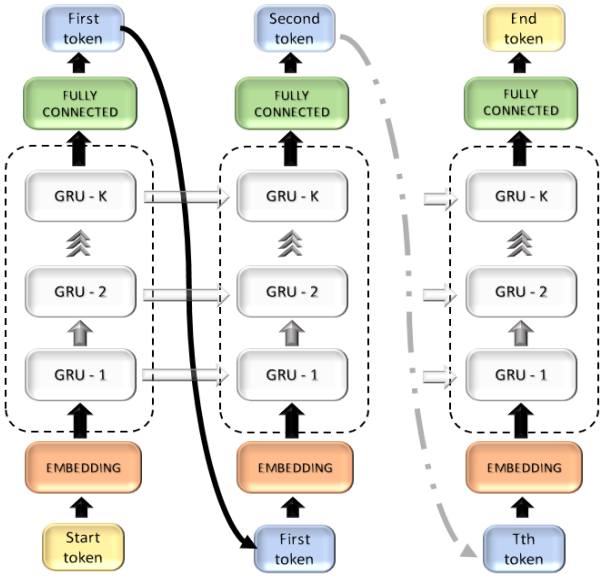


Figure 1. Multi-layer gated recurrent unit

Init-inject architecture: The hidden state vector of GRU is initialized with the same-sized image feature vector (Tanti et al., 2018), and the embedding vector is fed to the GRU as an input vector. The probability calculation is represented in the init-inject architecture as

$$p_{t+1} = \big(GRU(x_t), CNN(I)\big), \quad t \in \{0, 1, \dots, N-1\} \quad (7)$$

where *GRU* is the abbreviation of all processes in the Eqs. 3 - 6 and *CNN* is the abbreviation of the process of the encoder.

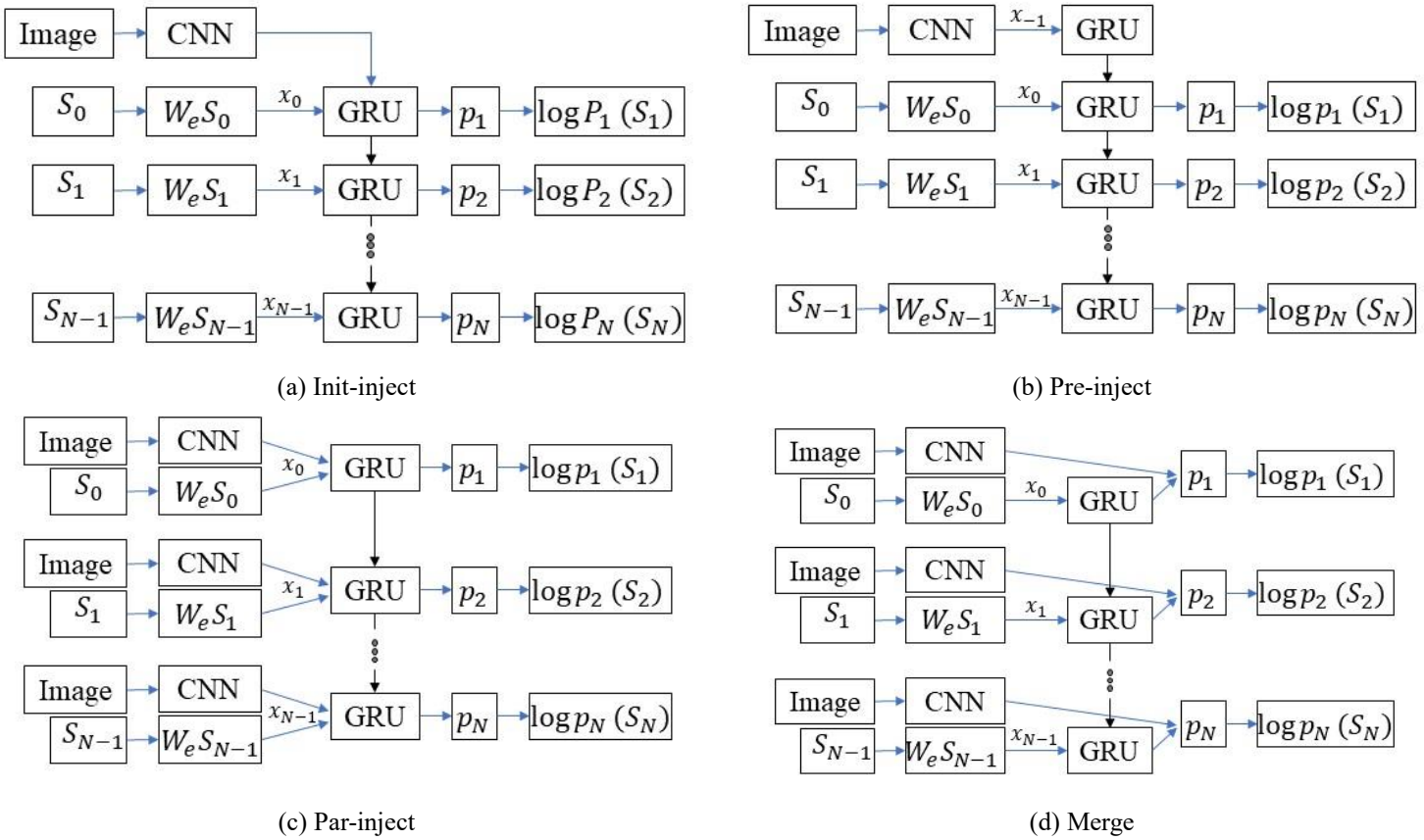
Pre-inject architecture: The image feature vector is utilized as the first input vector of GRU at $t = -1$, whereas the embedding vectors feed the GRU for the next step (Tanti et al., 2018). The image feature vector can be considered as the first word of the sequence. The process in the pre-inject architecture is represented as (Vinyals et al., 2015)

$$x_{t-1} = CNN(I) \quad (8)$$

$$x_t = W_e S_t \;\; t \in \{0, 1, \dots, N-1\} \quad (9)$$

$$p_{t+1} = GRU(x_t), \;\; t \in \{0, 1, \dots, N-1\} \quad (10)$$

where W_e denotes the weight of the embedding layer.



(a) Init-inject (b) Pre-inject

(c) Par-inject (d) Merge

Figure 2. An illustration of the four different architectures incorporating the visual and linguistic feature vector representations into the decoder. The image feature vector is used as an initial state of GRU in (a). The image feature vector is used as the first input in (b). The concatenated image feature vector and embedding vector are incorporated into the GRU in (c). GRU is fed only with an embedding vector, then GRU output is combined with the image vector in (d).

Par-inject architecture: The image and embedding vector are concatenated as a single input before incorporating into the GRU (Tanti et al., 2018). The probability calculation is represented in the par-inject architecture as

$$p_{t+1} = GRU\left(concat\left(x_t, CNN(I)\right)\right), \quad t \in \{0, 1, \dots, N-1\} \tag{11}$$

where *concat* represents the concatenation that joins existing vectors together.

Merge architecture: GRU takes only the embedding vector that handles linguistic features in this architecture, whereas the image feature vector is fed into the architecture after the GRU processes the linguistic features (Tanti et al., 2018). The image feature vector and the output vector of the GRU are merged into a single vector to calculate the probability. The probability calculation of the merge architecture is given as

$$p_{t+1} = concat\left(GRU(x_t), CNN(I)\right), \quad t \in \{0, 1, \dots, N-1\} \tag{12}$$

The general structure of the four architectures is illustrated in Figure 2.

3. Experimental Evaluations

This section presents the dataset, performance metrics and the evaluation of feature-injection architectures on multi-layer GRU based decoder with or without the FC layer.

3.1. Dataset and Performance Metrics

In order to evaluate feature-injection architectures, a dataset including a variety of images with reference captions is required. Flickr8k (Hodosh, Young, & Hockenmaier, 2013), Flickr30k (Young, Lai, Hodosh, & Hockenmaier, 2014), VizWiz-Captions (Gurari et al., 2018) and MSCOCO (T.-Y. Lin et al., 2014) are publicly available image captioning datasets. Flickr8k and Flickr30k include 8000 and 31783 images with five reference captions focused on people and objects involved in specific events and activities. The VizWiz- Captions dataset consists of 23431 training, 7750 validation and 8000 test images taken by blind people, paired with five reference captions. MSCOCO is a relatively large dataset containing 118287 training and 5000 validation images, each annotated with at least five reference captions. As a result, the MSCOCO dataset was employed, which is the most commonly used in captioning studies because it comprises all of the images in the Flickr datasets and offers a wide diversity of contents.

The benchmark of the feature-injection architectures was built with several metrics such as BLEU-n (n = 1, 2, 3, 4), ROUGE-L, SPICE, METEOR and CIDEr. BLEU was initially developed to evaluate the machine translation system, which counts the number of co-occurrence n-grams in the system and reference captions. METEOR is also used to evaluate machine translation, which considers the accuracy, recall rate, and F-value of the entire corpus. ROUGE-L is an automated text summary evaluation metric based on the longest subsequence at the sentence level. SPICE is a semantic metric for image captioning that evaluates by considering the objects, attributes, and relationships in the generated caption.

Table 1. Performance metrics results of different feature-injection architecture-based image captioning systems on MSCOCO dataset.

Decoder Design	Decoder	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	SPICE	METEOR	CIDEr
single-layer GRU	Init-inject	0.6232	0.4273	0.2828	0.1869	0.4545	0.1280	0.1993	0.6001
	Par-inject	**0.6336**	**0.4430**	**0.2971**	**0.1957**	**0.4621**	**0.1295**	**0.2008**	**0.6210**
	Pre-inject	0.6235	0.4273	0.2849	0.1893	0.4549	0.1240	0.1961	0.5821
	Merge	0.5923	0.3930	0.2547	0.1665	0.4279	0.1170	0.1842	0.5102
single-layer GRU with FC layer	Init-inject	0.5934	0.3950	0.2584	0.1696	0.4316	0.1140	0.1868	0.5131
	Par-inject	0.6304	**0.4348**	**0.2910**	**0.1932**	**0.4553**	**0.1251**	**0.1961**	**0.5903**
	Pre-inject	**0.6325**	0.4331	0.2883	0.1907	0.4540	0.1232	0.1948	0.5848
	Merge	0.5857	0.3823	0.2443	0.1581	0.4226	0.1077	0.1809	0.4953
3-layer GRU	Init-inject	**0.6379**	**0.4476**	**0.3038**	**0.2045**	**0.4640**	**0.1349**	**0.2067**	**0.6524**
	Par-inject	0.6200	0.4247	0.2850	0.1898	0.4517	0.1221	0.1938	0.5693
	Pre-inject	0.6169	0.4213	0.2814	0.1868	0.4496	0.1193	0.1908	0.5522
	Merge	0.5898	0.3904	0.2517	0.1623	0.4261	0.1164	0.1833	0.4988
3-layer GRU with FC layer	Init-inject	0.5618	0.3592	0.2221	0.1387	0.4090	0.0910	0.1637	0.3830
	Par-inject	**0.6240**	**0.4290**	**0.2881**	**0.1933**	**0.4538**	**0.1218**	**0.1945**	**0.5815**
	Pre-inject	0.6177	0.4222	0.2825	0.1885	0.4511	0.1188	0.1910	0.5609
	Merge	0.5903	0.3847	0.2463	0.1595	0.4236	0.1077	0.1802	0.4882

CIDEr is a customized metric for an image captioning system to evaluate the captions using sentence similarity to capture the notions of grammatically, saliency, and accuracy. Unlike CIDEr and SPICE, other metrics based on the ranking of captions (Young et al., 2014) and cannot evaluate novel image descriptions.

3.2. Results and Discussion

The feature-injection architectures have been examined under different designs of the multi-layer GRU based decoder and an FC layer to observe the effects in caption generation. All architectures were evaluated with BLEU-n, ROUGE-L, SPICE, METEOR, and CIDEr metrics on MSCOCO dataset. An FC layer is utilized before GRU for reducing the size of the image feature vector from 2048 to 128 as it leads further improvement in par-inject architecture (Tanti et al., 2018). The negative log-likelihood function and Adam optimization algorithm are applied with the following training parameters: 50 epochs, batches of 128 image caption pairs and a hidden state size of 128.

The experimental results in Table 1 are mainly evaluated based on CIDEr due to its better correlation with human assessment compared to other metrics. Extensive experiments indicate that inject-based architectures mostly outperform the merge architecture in terms of all performance metrics. The par-inject architecture performs better than the other architectures for a single-layer GRU regardless of the FC layer. The connection of the FC layer to the par-inject does not lead to any improvement in the single-layer GRU, however, the improvement is clear in the 3-layer GRU. On the contrary, the FC layer does not have a direct impact on the other architectures. The performance of the init-inject architecture is enhanced with the increase of the layer size contrary to others. Among all the designs, the highest performance is obtained by 3-layer GRU under the init-inject architecture which incorporates the high-level visual information without the FC layer before the GRU. Table 2 shows the ground truth (reference) and generated captions on samples from MSCOCO for each decoder design. These results indicate that the generated captions are coherent with the CIDEr scores as more meaningful captions are obtained.

4. Conclusions

In this study, a benchmark of feature-injection architectures that incorporate visual information into the decoder is reported. These architectures are based on a CNN that encodes an image into a feature vector, followed by a GRU based decoder that generates the corresponding caption of the image. These architectures are tested with multi-layer GRU, and an FC layer to see their contributions to captioning performance. Extensive evaluations of architectures on the MSCOCO dataset demonstrate that the init-inject architecture with multi-layer GRU design offers promising performance compared to the other architectures. In the future study, the effect of hyper-parameters on the generating captions will be investigated to provide a tuning strategy for respective architectures.

5. Acknowledge

This research was supported by the Scientific and Technological Research Council of Turkey (TUBITAK)-British Council (The Newton-Katip Celebi Fund Institutional Links, Turkey-UK project: 120N995) and TUBITAK 2209-B Industry Oriented Research Project Support Programme for Undergraduate Students with project no: 1139B412000694.

Table 2. Examples of ground truth and generated captions of images selected from the MSCOCO validation set.

Decoder Design	MSCOCO Images	Reference Captions	Generated Captions
Single-layer GRU		• A baseball game is in action as a batter swings. • A baseball game with a batter ready to swing. • A batter, catcher and umpire in a baseball game. • A photo of a person being taken in this picture. • A baseball player holding a bat while standing on a field.	**Init-inject:** Man swinging a baseball bat on a field. **Par-inject:** A baseball player swinging a bat at a game. **Pre-inject:** A baseball player is getting ready to hit a ball. **Merge:** A baseball player is getting ready to hit the ball.
Single-layer GRU with FC layer		• A rain covered terrain after a night of rain. • A street at night time with many different lights. • A bright city street with a stop light and a big christmas tree. • An empty street at night with lots of lights in the background. • A red traffic light at night next to a christmas tree.	**Init-inject:** A night time view of a city street with a church in the distance. **Par-inject:** A night view of a city street with lit up tower. **Pre-inject:** A street with a lot of cars and a large building. **Merge:** A street light at night with lights on.
3-layer GRU		• A dog that is sitting down in a backseat. • An adorable brown and white dog hanging it's head out of a window. • A dog looking out the window as seen through a mirror. • A dog has its head hanging out of a window. • The reflection of a dogs head out of a car window in one of the cars wing mirrors.	**Init-inject:** A dog is sitting on a car with his head sticking out the window. **Par-inject:** A dog is standing in the car. **Pre-inject:** A dog that is sitting on a car. **Merge:** A man in a car is seen through the mirror.
3-layer GRU with FC layer		• A man walking next to a snowy hill. • A cross country skier traveling down a slight slope. • A man with skis and ski poles is standing next to a hill covered in snow. • The elderly man on skis is making his way down the edge of the snow-covered road. • A person on skis riding down a snowy slope.	**Init-inject:** A group of people who are standing in the snow. **Par-inject:** A person is walking down a hill with a pair of skis. **Pre-inject:** A man is skiing in a snowy field. **Merge:** A man is standing in a field with a pair of skis.
Comparison of best decoder architectures		• Water traffic along the thames by big ben. • A barge floating down a river with the skyline in the background. • The enveloping of an outside town in the picture. • Tall building sitting on the rivers edge and a barge. • A castle and the big ben clocktower next to a river.	**Par-inject:** A view of a big clock tower in London. **(Single-layer GRU)** **Par-inject:** A big ben clock tower towering over a city. **(Single-layer with FC layer)** **Init-inject:** A view of a river with boats and a clock tower in the background. **(3-layer GRU)** **Par-inject:** A large building with a clock tower on it. **(3-layer with FC layer)**

References

Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). *Spice: Semantic propositional image caption evaluation.* Paper presented at the European Conference on Computer Vision.

Banerjee, S., & Lavie, A. (2005). *METEOR: An automatic metric for MT evaluation with improved correlation with human judgments.* Paper presented at the Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization.

Baran, M., Moral, Ö. T., & Kılıç, V. (2021). Akıllı Telefonlar için Birleştirme Modeli Tabanlı Görüntü Altyazılama. *Avrupa Bilim ve Teknoloji Dergisi*(26), 191-196.

Çaylı, Ö., Makav, B., Kılıç, V., & Onan, A. (2020). *Mobile Application Based Automatic Caption Generation for Visually Impaired.* Paper presented at the International Conference on Intelligent and Fuzzy Systems.

Chang, S.-F. (1995). *Compressed-domain techniques for image/video indexing and manipulation.* Paper presented at the Proceedings., International Conference on Image Processing.

Chiarella, D., Yarbrough, J., & Jackson, C. A.-L. (2020). Using alt text to make science Twitter more accessible for people with visual impairments. *Nature Communications, 11*(1), 1-3.

Chollet, F. (2017). *Xception: Deep learning with depthwise separable convolutions.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:.*

Devlin, J., Cheng, H., Fang, H., Gupta, S., Deng, L., He, X., . . . Mitchell, M. J. a. p. a. (2015). Language models for image captioning: The quirks and what works.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). *Long-term recurrent convolutional networks for visual recognition and description.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Gao, Y., & Glowacka, D. (2016). *Deep gate recurrent neural network.* Paper presented at the Asian conference on machine learning.

Gers, F. A., & Schmidhuber, E. (2001). LSTM recurrent networks learn simple context-free and context-sensitive languages. *IEEE Transactions on Neural Networks Learning Systems, 12*(6), 1333-1340.

Gurari, D., Li, Q., Stangl, A. J., Guo, A., Lin, C., Grauman, K., . . . Bigham, J. P. (2018). *Vizwiz grand challenge: Answering visual questions from blind people.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Hochreiter, S., & Schmidhuber, J. J. N. c. (1997). Long short-term memory. *9*(8), 1735-1780.

Hodosh, M., Young, P., & Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research, 47*, 853-899.

Keskin, R., Moral, Ö. T., Kılıç, V., & Onan, A. (2021). *Multi-GRU Based Automated Image Captioning for Smartphones.* Paper presented at the 2021 29th Signal Processing and Communications Applications Conference (SIU).

Kılıç, V. (2021). Deep Gated Recurrent Unit for Smartphone-Based Image Captioning. *Sakarya University Journal of Computer Information Sciences, 4*(2), 181-191.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., . . . Berg, T. L. (2013). Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis Machine Intelligence, 35*(12), 2891-2903.

Lin, C.-Y. (2004). *Rouge: A package for automatic evaluation of summaries.* Paper presented at the Text Summarization Branches Out.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. L. (2014). *Microsoft coco: Common objects in context.* Paper presented at the European Conference on Computer Vision.

Liu, S., Zhu, Z., Ye, N., Guadarrama, S., & Murphy, K. (2016). Optimization of image description metrics using policy gradient methods.

Liu, X., Xu, Q., & Wang, N. (2019). A survey on deep neural network-based image captioning. *The Visual Computer, 35*(3), 445-470.

Makav, B., & Kılıç, V. (2019a). *A new image captioning approach for visually impaired people.* Paper presented at the 2019 11th International Conference on Electrical and Electronics Engineering (ELECO).

Makav, B., & Kılıç, V. (2019b). *Smartphone-based image captioning for visually and hearing impaired.* Paper presented at the 2019 11th International Conference on Electrical and Electronics Engineering (ELECO).

Mao, J., Wei, X., Yang, Y., Wang, J., Huang, Z., & Yuille, A. L. (2015). *Learning like a child: Fast novel visual concept learning from sentence descriptions of images.* Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.

Nina, O., & Rodriguez, A. (2015). *Simplified LSTM unit and search space probability exploration for image description.* Paper presented at the 2015 10th International Conference on Information, Communications and Signal Processing (ICICS).

Ordonez, V., Kulkarni, G., & Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. *Advances in Neural Information Processing Systems, 24*, 1143-1151.

Ouyang, H., Zeng, J., Li, Y., & Luo, S. J. P. (2020). Fault detection and identification of blast furnace ironmaking process using the gated recurrent unit network. *8*(4), 391.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). *Bleu: a method for automatic evaluation of machine translation.* Paper presented at the Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). *Rethinking the inception architecture for computer vision.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Tanti, M., Gatt, A., & Camilleri, K. P. (2018). Where to put the image in an image caption generator. *Natural Language Engineering, 24*(3), 467-489.

Tao, Y., Wang, X., Sánchez, R.-V., Yang, S., & Bai, Y. (2019). Spur gear fault diagnosis using a multilayer gated recurrent unit approach with vibration signal. *IEEE Access, 7*, 56880-56889.

Targ, S., Almeida, D., & Lyman, K. (2016). Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:.08029*.

Vedantam, R., Lawrence Zitnick, C., & Parikh, D. (2015). *Cider: Consensus-based image description evaluation.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). *Show and tell: A neural image caption generator.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2016). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE Transactions on Pattern Analysis Machine Intelligence, 39*(4), 652-663.

Yao, T., Pan, Y., Li, Y., Qiu, Z., & Mei, T. (2017, 22-29 Oct. 2017). *Boosting Image Captioning with Attributes.* Paper presented at the 2017 IEEE International Conference on Computer Vision (ICCV).

Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics, 2*, 67-78.